WHAT IS CLAIMED IS:

1. A method for validating performance and functionality of a processor, comprising the steps of:

executing a program on a high level simulator of said processor;

establishing a plurality of checkpoints;

saving state data at each of said checkpoints; and

running said program on a plurality of low level simulators of said processor in parallel, starting each of said low level simulators at a corresponding checkpoint with corresponding state data associated with said corresponding checkpoint.

2. The method of claim 1 wherein said checkpoints divide said program into code fragments of determined lengths.

3. The method of claim 1 wherein said checkpoints divide said program into code fragments of random lengths.

4. The method of claim 1 wherein said state data comprises:

program counter contents of set processor;

register contents of said processor;

cache memory contents of said processor;

main memory contents of said processor; and

branch prediction contents of said processor.

5. The method of claim 1 wherein said processor is one of (a) a microprocessor, (b) a digital signal processor, (c) an input/output (I/O) controller, and (d) a network processor.

6. The method of claim 1 wherein said high level simulator is one of (a) an instruction accurate simulator (IAS) of said processor and (b) a cycle accurate simulator (CAS) of said processor.

7. The method of claim 1 wherein each of said low level simulators is a register transfer level (RTL) model of said processor, written as one of (a) a VHDL model of said processor and (b) a Verilog model of said processor.

8. The method of claim 1 wherein said running step further comprises the steps of:

loading each of said low level simulators with said program;

initializing each of said low level simulators at said corresponding checkpoint with said corresponding state data associated with said corresponding checkpoint; and

executing said program on said low level simulator up to a certain point in said program.

9. The method of claim 8 wherein said certain point is one of (a) a next checkpoint immediately following said corresponding checkpoint, (b) a point in said program a random length after said corresponding checkpoint, and (c) a point after said corresponding checkpoint.

10. The method of claim 1 wherein said running step further comprises generating one of (a) functional data of said processor and (b) performance data of said processor.

11. A computer readable media having stored thereon a program for validation of performance and functionality of a processor, comprising computer readable instructions for:

executing a program on a high level simulator of said processor;

establishing a plurality of checkpoints;

saving state data at each of said checkpoints; and

running said program on a plurality of low level simulators of said processor in parallel, starting each of said low level simulators at a corresponding checkpoint with corresponding state data associated with said corresponding checkpoint.

12. The computer readable media of claim 11 wherein said checkpoints divide said program into code fragments of determined lengths.

13. The computer readable media of claim 11 wherein said checkpoints divide said program into code fragments of random lengths.

14. The computer readable media of claim 11 wherein said computer readable instructions for running said program on a plurality of low level simulators of said processor in parallel, further comprises computer readable instructions for:

loading each of said low level simulators with said program;

initializing each of said low level simulators at said corresponding checkpoint with said corresponding state data associated with said corresponding checkpoint; and

executing said program on said low level simulator up to a certain point in said program.

15. A computer system for validating performance and functionality of a processor, comprising:

a first computer programmed with a high level simulator of said processor and configured to run a program;

a memory for storing checkpoints and state data at each of said checkpoints; and

a plurality of second computers programmed with low level simulators of said processor and configured to run said program in parallel, each of said second computers starting at a different one of said checkpoints with a corresponding state data.